Liaison 5.5

User Manual

Schrödinger Press

June 2009

Contents

Document Conventions

In addition to the use of italics for names of documents, the font conventions that are used in this document are summarized in the table below.

Font	Example	Use
Sans serif	Project Table	Names of GUI features, such as panels, menus, menu items, buttons, and labels
Monospace	`$SCHRODINGER/maestro`	File names, directory names, commands, environment variables, and screen output
Italic	*filename*	Text that the user must replace with a value
Sans serif uppercase	CTRL+H	Keyboard keys

Links to other locations in the current document or to other PDF documents are colored like this: Document Conventions.

In descriptions of command syntax, the following UNIX conventions are used: braces `{ }` enclose a choice of required items, square brackets `[ ]` enclose optional items, and the bar symbol | separates items in a list from which one item must be chosen. Lines of command syntax that wrap should be interpreted as a single command.

File name, path, and environment variable syntax is generally given with the UNIX conventions. To obtain the Windows conventions, replace the forward slash / with the backslash \ in path or directory names, and replace the $ at the beginning of an environment variable with a % at each end. For example, `$SCHRODINGER/maestro` becomes `%SCHRODINGER%\maestro`.

In this document, to *type* text means to type the required text in the specified location, and to *enter* text means to type the required text, then press the ENTER key.

References to literature sources are given in square brackets, like this: [10].

Introduction

1.1 About Liaison

Liaison predicts ligand-receptor binding affinities using a linear interaction approximation (LIA) model that has been fitted to a set of known binding free energies. For each ligand in the training set, Liaison runs molecular mechanics (MM) simulations of the ligand-receptor complex at both endpoints of the binding process, bound ligand and free ligand. The simulation data and empirical binding affinities are analyzed to generate the Liaison parameters: α, β, and γ. These parameters are subsequently used to predict binding energies for other ligands with the same receptor. Liaison simulations use a continuum solvation model to shorten sampling times and speed convergence.

Liaison is run primarily from the Maestro graphical user interface. A tutorial in using Liaison from Maestro appears in Chapter 2. Liaison can also be run from the command line, as described in Chapter 5. Utilities and scripts are run from the command line. Liaison technical notes, background, and references are provided in Chapter 5.

Maestro is Schrödinger's powerful, unified, multi-platform graphical user interface (GUI). It is designed to simplify modeling tasks, such as molecule building and data analysis, and also to facilitate the set up and submission of jobs to Schrödinger's computational programs. The main Maestro features include a project-based data management facility, a scripting language for automating large or repetitive tasks, a wide range of useful display options, a comprehensive molecular builder, and surfacing and entry plotting facilities. For detailed information about the Maestro interface, see the Maestro online help or the *Maestro User Manual*.

Protein Preparation is strongly recommended for protein and protein-ligand complex PDB structures to be used in Liaison. In most cases, this can be performed in Maestro, using the Protein Preparation Wizard panel on the Workflows menu. Protein and ligand preparation are described in Chapter 3.

The **Impact** computational program runs the MM calculations for Liaison simulations, which can be carried out using molecular dynamics (MD), hybrid Monte Carlo (HMC), or energy minimization. Impact uses an OPLS-AA force field. Impact calculations can also be run independently of Liaison. For more information, see the *Impact User Manual* and the *Impact Command Reference Manual*.

The Strike statistical analysis package is used for the fitting and prediction analysis tasks of Liaison. For more information, see the *Strike User Manual*.

1.2 Liaison Binding Energy Models

1.2.1 LIA Model Equation

A Liaison simulation combines a molecular-mechanics calculation with experimental data to build a model scoring function used to correlate or to predict ligand-protein binding free energies. The assumption used is that the binding energy can be approximated by comparing the energy of the bound complex with the energy of the free ligand-receptor system. A method of this type is called a Linear Response Method (LRM), a Linear Interaction Approximation (LIA), or a Linear Interaction Energy (LIE) method.

A novel feature of Liaison is that the simulation takes place in implicit (continuum) rather than explicit solvent—hence the name Liaison, for Linear Interaction Approximation in Implicit SOlvatioN. The explicit-solvent version of the methodology was first suggested by Aqvist (Hansson, T.; Aqvist, *J. Protein Eng.* **1995,** *8,* 1137-1145), based on approximating the charging integral in the free-energy-perturbation formula with a mean-value approach, in which the integral is represented as half the sum of the values at the endpoints, namely the free and bound states of the ligand. The empirical relationship used by Liaison is shown below:

$$\Delta G = \alpha\,(\langle U^{b}_{vdw}\rangle - \langle U^{f}_{vdw}\rangle) + \beta\,(\langle U^{b}_{elec}\rangle - \langle U^{f}_{elec}\rangle) + \gamma\,(\langle U^{b}_{cav}\rangle - \langle U^{f}_{cav}\rangle)$$

Here < > represents the ensemble average, *b* represents the bound form of the ligand, *f* represents the free form of the ligand, and α, β, and γ are the coefficients. U_{vdw}, U_{elec}, and U_{cav} are the van der Waals, electrostatic, and cavity energy terms in the Surface Generalized Born (SGB) continuum solvent model. The cavity energy term, U_{cav}, is proportional to the exposed surface area of the ligand. Thus, the difference:

$$\langle U^{b}_{cav}\rangle - \langle U^{f}_{cav}\rangle$$

measures the surface area lost by contact with the receptor. The net electrostatic interaction-energy in continuum solvent is given by:

$$U_{elec} = U_{coul} + 2\,U_{rxnf}$$

where U_{coul} is the Coulomb interaction energy and U_{rxnf} is the SGB-solvent reaction-field energy. (The factor of 2 compensates for the division by 2 made in the definition of the reaction-field free energy.)

In most applications, the coefficients α, β, and γ are determined empirically by fitting to the experimentally determined free energies of binding for a training set of ligands. In such applications, Liaison's simulation task is used to calculate the values of U_{vdw}, U_{elec}, and U_{cav} for the bound (complexed) and unbound (free) states of the training-set ligands, and its analysis task is used to derive values for the α, β, and γ fitting coefficients. The fitted equation can then be used to predict the binding affinities of additional ligands. In the current version of Liaison, a

constant term is added to ΔG in the fitting process, and is adjusted during the fit. This corresponds to an extension of the strict linear response model.

1.2.2 LiaisonScore Model Equation

Liaison also calculates a scoring function similar to GlideScore over the course of the LRM simulation. This scoring function, previously called "GlideScore in Liaison," is called LiaisonScore in Liaison 5.5. The average LiaisonScore can then be used to predict binding energies using the alternate model:

$$\Delta G = a(\langle LiaisonScore \rangle) + b$$

where a is the LiaisonScore coefficient and b is a constant.

1.3 Running Schrödinger Software

To run any Schrödinger program on a UNIX platform, or start a Schrödinger job on a remote host from a UNIX platform, you must first set the `SCHRODINGER` environment variable to the installation directory for your Schrödinger software. To set this variable, enter the following command at a shell prompt:

csh/tcsh: `setenv SCHRODINGER` *installation-directory*

bash/ksh: `export SCHRODINGER=`*installation-directory*

Once you have set the `SCHRODINGER` environment variable, you can start Maestro with the following command:

```
$SCHRODINGER/maestro &
```

It is usually a good idea to change to the desired working directory before starting Maestro. This directory then becomes Maestro's working directory. For more information on starting Maestro, including starting Maestro on a Windows platform, see Section 2.1 of the *Maestro User Manual*.

1.4 Citing Liaison in Publications

The use of this product and its components should be acknowledged in publications as:

Liaison, version 5.5, Schrödinger, LLC, New York, NY, 2009; Strike, version 1.8, Schrödinger, LLC, New York, NY, 2009.

Liaison Tutorial

This chapter contains tutorial exercises to help you quickly become familiar with the functionality of Liaison using the Maestro interface. Liaison is used to simulate and predict binding affinities. It does so by generating for each protein-ligand complex the descriptors necessary to apply the LIA equation, the LiaScore, which is the GlideScore computed over discrete protein atom locations rather than over a gridded protein representation, and the LiaScore components which may be used to generate an ELR model. Models for the binding affinity are then created and applied from Liaison-generated descriptors via Strike. Thus, the Liaison process involves two steps, the simulation of binding in Liaison to generate a set of descriptors and the creation and application of binding affinity models from the Liaison descriptors with Strike.

The exercises in this chapter demonstrate:

- How to perform Liaison simulations on multiple ligands
- How to create a validated model for the α, β, and γ coefficients for the LIA equation using Strike
- How to generate and apply the results of Liaison simulations to predict binding affinities for novel ligands
- How to create and apply LiaScore and Extended Linear Response (ELR) models to predict binding affinities

You will use the Liaison panel to set up and run Liaison simulations and then use the Strike panels to create, validate, and apply binding affinity models from Strike.

To do these exercises you must have access to an installed version of Maestro 9.0, Liaison 5.5 and Strike 1.8. For installation instructions, see the *Installation Guide*.

2.1 Preparation

Before you start Maestro and begin the exercises, you must first create a local tutorial directory tree. Some exercises in this tutorial produce files that are needed in subsequent exercises. To allow you to begin at any exercise you choose, the `$SCHRODINGER/impact-v`*version*`/tutorial/liaison` directory contains copies of the relevant input and output files that will be generated as you perform the tutorial.

To create a local directory tree:

1. At a shell prompt, change to a directory in which you have write permissions.
2. Create a local base directory:

   `mkdir` *basedir*

   In the text, this directory is referred to as the base directory.
3. In the base directory, create a soft link to the `$SCHRODINGER/impact-v`*version*`/tutorial/liaison` directory by entering the following command:

   `ln -s $SCHRODINGER/impact-v`*version*`/tutorial/liaison .`

2.2 Starting Maestro

You do not need to start Maestro until you begin an exercise. If you have not started Maestro before, this section contains instructions.

To start Maestro:

1. Set the `SCHRODINGER` environment variable to the directory in which Maestro and Liaison are installed:

   csh/tcsh: `setenv SCHRODINGER` *installation_path*

   sh/bash/ksh: `export SCHRODINGER=`*installation_path*

2. Change to the desired working directory.

   `cd` *basedir*
3. Enter the command:

   `$SCHRODINGER/maestro &`

You are now ready to proceed with the exercises below.

2.3 Running the Liaison Simulations

In this exercise, starting with a receptor and a set of prepared ligands, you will set up and run a job that calculates Liaison descriptors. The descriptors are saved in a Maestro file and a CSV file, allowing you to use them in the creation, validation, and application of LIA, LiaScore and ELR models of binding affinity.

The Liaison calculations to be run in this exercise require about 1.5 hours of CPU time on a single 2.8 GHz Xeon processor, though by taking advantage of multiple processors this time

can be reduced sharply. The output files from the Liaison simulation have been prepared for this exercise so the tutorial may be completed whether you choose to run the Liaison simulations or not.

2.3.1 Setting Up the System

Before running Liaison on a series of ligand-receptor complexes, you must import the receptor and ligands and set the parameters for the Liaison simulations.

1. From the Applications menu in the main window, choose Liaison.

   The Liaison panel opens with the Systems tab displayed.

2. In the Specify structures section, ensure that Take complexes from a Maestro Pose Viewer file is selected.

3. Click Browse.

4. In the file selector, navigate to your *basedir*`/liaison` directory and import the file `1rt1_hept_analogs_pv.mae`.

The ligand-protein complexes to be simulated have now been specified.

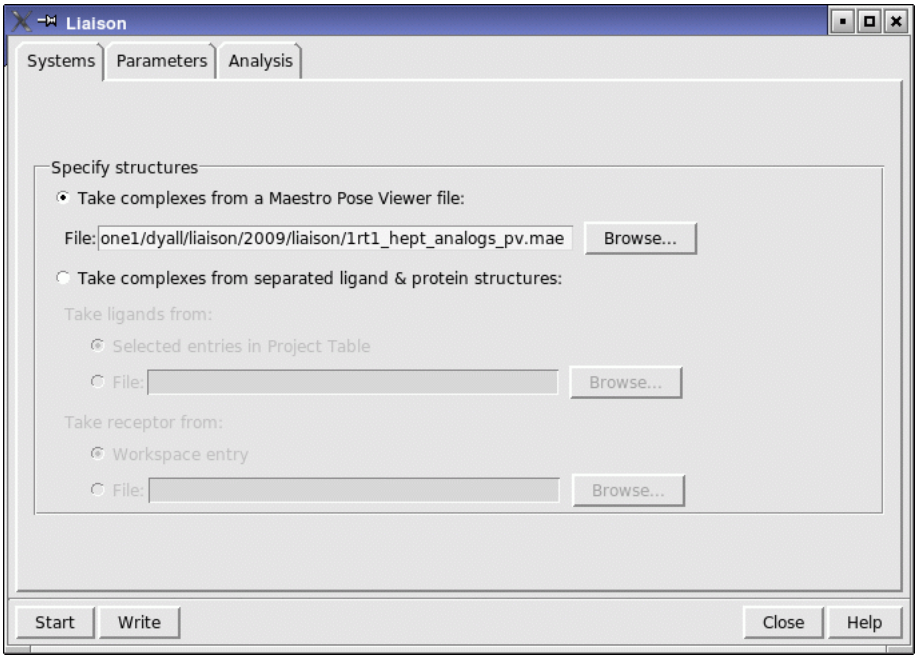


Figure 2.1. The Systems tab of the Liaison panel.

To prepare the Liaison simulation parameters:

5. In the Job options section of the Parameters tab, choose OPLS_2001 from the Force field option menu.
6. In the Specify restrained/frozen shells section, choose Huge from the option menu.

2.3.2 Starting and Monitoring the Liaison Job

The job takes approximately 1.5 hours on a 2.8 GHz Xeon processor. The Liaison simulations do not need to be run to continue with the tutorial. If you prefer, you may continue the tutorial starting with Section 2.4.

With the ligands and receptors defined and Liaison simulation parameters set, the Liaison simulation can be started.

1. Click Start.

   The Start dialog box opens.
2. Change the Job Name to `hept_analogs`.
3. Select the host where the Liaison simulations are to be run from the Host option menu.

   To run on the local machine ensure that `localhost` is selected. If the host is a multiprocessor machine, specify the number of available CPUs in the Use *N* Processors text box, otherwise leave its value at one.
4. Click Start.

   The Monitor panel opens with the File tab displayed.

While the `hept_analogs` job is in progress, the Status column in the Jobs tab for this job displays the text "running". When the job is complete, the status changes to "incorporated : finished". The `hept_analogs` job spawns a subprocess named `hept_analogs_sim`. This subprocess in turn spawns a job for each ligand-receptor complex.

Before the job is launched the following input files are written:

`hept_analogs.inp`	LSBD command input to run Liaison simulation
`hept_anlaogs-ligs.mae`	Ligand structure input for Liaison simulation
`hept_analogs-rec.mae`	Receptor structure input for Liaison simulation
`hept_analogs-lia_cons.mae`	Ligand structure from which restrained/frozen receptor atoms are determined

When the Liaison simulation finishes, the calculated Liaison results are incorporated into the Project Table along with the input ligand geometries, and the *basedir* directory will contain the following job output files:

`hept_analogs.log`	LSBD log summary file
`hept_analogs-final.mae`	Structure file of input ligand geometries with calculated Liaison results
`hept_analogs.csv`	Comma-separated value file with calculated Liaison results
`hept_analogs.tar.gz`	Compressed tarball containing the full set of Liaison results

If you want to stop working on the tutorial now, choose Close Project from the Project menu. If the project is a scratch project, you will be prompted to save it or delete it.

2.4 Generating and Validating an LIA Model of Binding Affinity

Before you begin the exercises shown below, you must first have created a local tutorial directory tree, as described in Section 2.1 on page 5. If you have not yet set up your tutorial directory tree, do so now, then proceed with the exercises.

A Strike license is required for these exercises.

2.4.1 Importing Liaison Results into Strike for Model Creation

You will be importing the Liaison descriptors that will be used to create the LIA model through the Liaison panel. The data could also be imported directly into the Project Table. The Liaison descriptors are stored in both the `hept_analog-final.mae` and `hept_analog.csv` files. For this tutorial you will import the data using the Maestro file, into a new project.

1. Choose New from the Project menu in the main window.

   A project selector is displayed.

2. Enter `liasim` in the Name text box, and click Save.

3. Click Open/Close Project Table on the main toolbar.

   The Project Table panel is displayed.

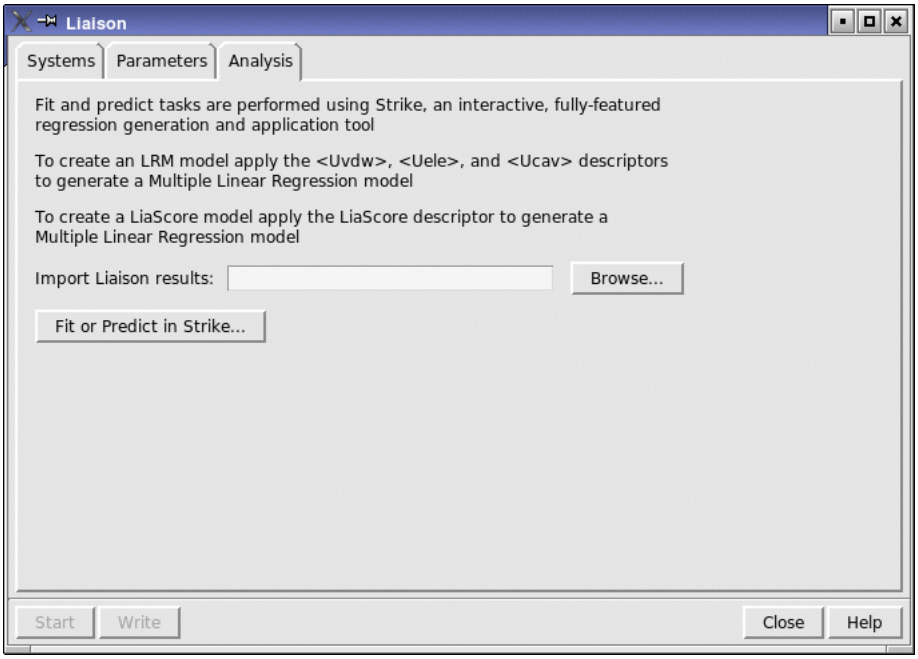


Figure 2.2. The Analysis tab of the Liaison panel.

4. Choose Liaison from the Applications menu in the main window.

   The Liaison panel opens with the Systems tab displayed.

5. In the Analysis tab, click Browse.

   A file selector opens.

6. Navigate to and import the file `hept_analogs-final.mae`.

   If you ran the Liaison simulation described in Section 2.3, this file is in your *basedir* directory. Otherwise, you can find a copy of it in your *basedir*`/liaison` directory.

7. In the Analysis tab, click Fit or Predict in Strike.

   The molecules and data are imported from `hept_analogs-final.mae` into the Project Table, and the Strike Build QSAR Model panel opens.

8. Close the Liaison panel.

2.4.2 Selecting Training Set Molecules

It is often important before model generation to separate available data into training and test sets. The training set is used to train a model to predict binding affinities. The resultant model is then validated by applying it to the prediction of binding affinities for molecules in the test set. The current data set has been separated into training and test sets as indicated in the Set property. Typically this separation is done with the Random option in the Select menu of the Project Table panel.

Strike creates models using selected entries in the Project Table. Since we are to generate an LIA model using only molecules in the training set, its important to ensure only training set molecules are selected in the Project Table:

1. In the Project Table panel, choose Select > Only.

   The Entry Selection panel opens. In the Properties tab there is a list of properties stored in the Project Table.

2. Select Set from the list in the Properties tab.
3. Select Matches and enter `Train` in the text field.
4. Click Add.

   The ESL text box is updated with the chosen matching condition.

5. Click OK.

   The Entry Selection panel closes, and in the Project Table only the molecules in the training set are selected.

2.4.3 Creating the LIA Model to Predict Binding Affinities

The experimental binding affinities given in the property Activity (kcal/mol) will be used for the response or dependent variable from which a linear model will be created. The LIA equation is outlined in detail in the Liaison Manual and it estimates binding affinities using $<U_{ele}>$, $<U_{vdw}>$, and $<U_{cav}>$ terms from Liaison.

In this exercise, you will create an LIA model for the twelve training set molecules you selected in the previous exercise. The Build QSAR Model panel should already be open from the exercise in Section 2.4.1. If not, choose Build QSAR Model from the Strike submenu of the Applications menu in the main window.

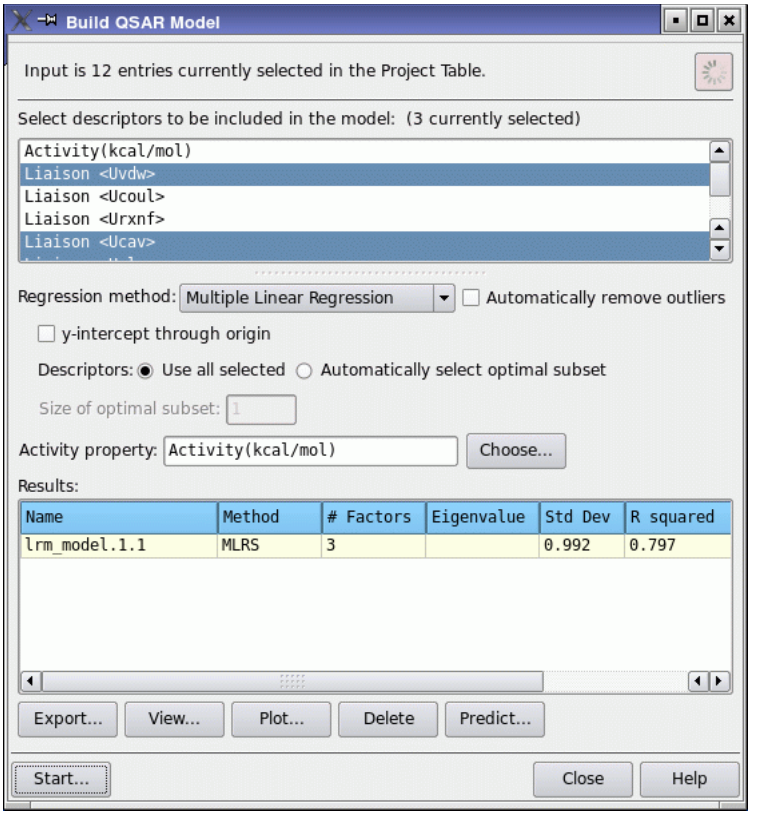


Figure 2.3. The Strike Build QSAR Model panel.

1. In the Select descriptors to be included in the model table, select Liaison <Uele>, Liaison <Uvdw>, and Liaison <Ucav>.

   Use control-click to select the second and third of these descriptors. These are the terms that are used for an LIA model, and will be the independent variables in the model.

2. From the Regression method option menu, choose Multiple Linear Regression.

3. Ensure that Use all selected is selected.

4. Click Choose, to the right of the Activity Property text box.

   The Choose Activity Property dialog box opens.

5. Select the Activity (kcal/mol) property and click OK.

   This property is the response or dependent variable which a model will be created to predict.

6. Click Start.

   A Start dialog box opens.

7. Enter `lrm_model` as the job name.

8. Click Start to begin the Strike job.

The Strike job should finish in a matter of seconds and the predicted binding affinities are incorporated in the Project Table as Predicted Activity1.1. In the Build QSAR Model panel, a row with name lrm_model.1.1 is added to the Results table that corresponds to the LIA model.

2.4.4 Analyzing the LIA Binding Affinity Model

Once the LIA model has been created, it must be analyzed to see if it makes intuitive sense and possesses predictive power that does not arise by chance. From the Build QSAR Model panel we will view a plot of predicted versus experimental activities, analyze the fundamentals of the model, and assess its predictive power prior to making predictions on test set molecules.

The basics of each model are displayed in the Results table in the Build QSAR Model panel and provide an at-a-glance overview of a model. For a multiple linear regression (MLR) model these include the standard deviation, R^2, F-statistic, and P-value. For the purposes of this tutorial we are interested in models with an R^2 greater than 0.6, a standard deviation lower than 1 log unit, and a P-value less than 0.05. For an LIA model to make intuitive sense the α, β, and γ coefficients calculated when using the OPLS_2001 force-field should all be positive. For OPLS_2005 the cavity term is calculated in a different fashion, and the gamma coefficient does not need to be positive for the LIA model to make intuitive sense. For further information on the fundamental metrics of an MLR model see the *Strike User Manual*.

Next, you will view all the available information on the LIA model.

1. Click View in the Build QSAR Model panel.

   The View QSAR Model panel opens. This panel displays the Strike output from LIA model creation. From this display all the information on the model is available, from basic regression statistics to validation tests such as the results of the leave-group-out and dependent variable randomization testing results.

2.4.5 Predicting Binding Affinities with the LIA Model

Once the LIA model has been analyzed and found to be suitable, it may be applied to predict binding affinities for molecules in the test set. In a similar fashion the LIA model may be applied to predict binding affinities for any molecule for which the LIA descriptors have been calculated through Liaison. What is essential is that the molecules must be imported into the Project Table where they can be acted upon by Strike.

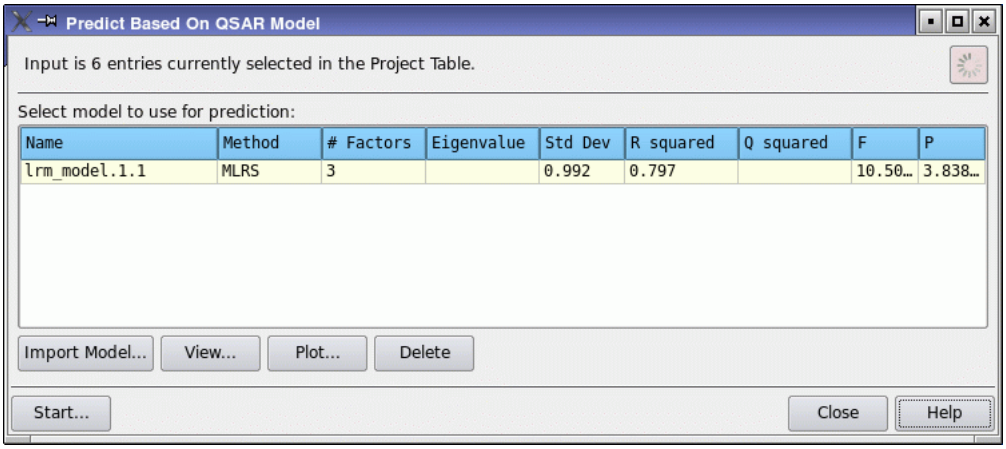


Figure 2.4. The Strike Predict based on QSAR model panel.

First the training set molecules must be selected in the Project Table. If you have changed the selection in the Project Table, repeat the exercise in Section 2.4.2, using `Test` instead of `Train` as the text to match, and skip to Step 2.

1. In the Project Table panel, choose Select > Invert.

   The six test set molecules are selected in the Project Table.

2. In the Build QSAR Model panel, click Predict.

   The Predict based on QSAR model panel opens, and the Build QSAR Model panel closes.

3. In the Select model to use for prediction table ensure that the LIA model is selected.

   This should be the model named lrm_model.1.1.

4. Click Start.

   A Start dialog box opens.

5. Enter `lrm_pred` as the job name.

6. Choose a host, then click Start to begin the Strike job.

The Strike job should finish in a matter of seconds, and the predicted binding affinities from the LIA model are incorporated into the Project Table as Activity (kcal/mol) Strike Prediction.

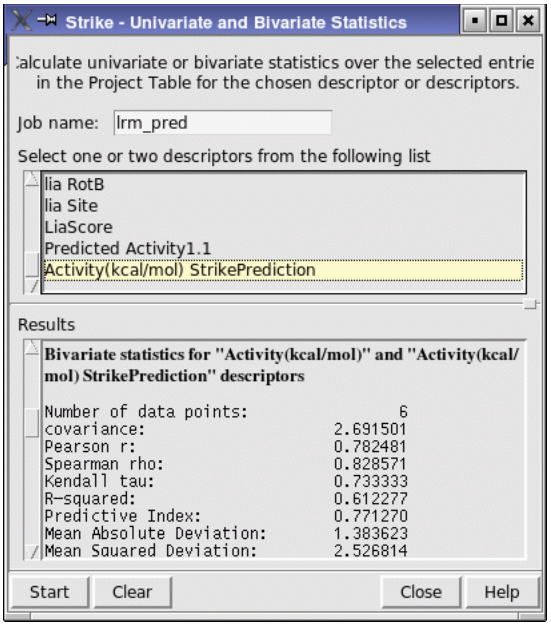


Figure 2.5. The Strike – Univariate and Bivariate statistics panel.

2.4.6 Analyze LIA Binding Affinity Predictions for the Test Set

We are interested in viewing how well the LIA model predicted binding affinities for molecules not included in the training set. Qualitatively the predicted and experimental binding affinities may be plotted using the plotting facilities of Maestro. Quantitatively, the correlation between the predicted and experimental binding affinities is given in the R^2 statistic. In this exercise you will generate an R^2 statistic for the test set molecules showing the correlation between the predicted and experimental binding affinities.

1. Ensure that only the six test set molecules are selected in the Project Table.

   This should be the case from the previous exercise unless you have changed the selection.

2. Choose Applications > Strike > Statistics in the main window.

   The Strike – Univariate and Bivariate statistics panel opens.

3. Enter `lrm_pred` as the job name.

4. Select Activity (kcal/mol) and Activity(kcal/mol) StrikePrediction to analyze the correlation between these two properties.

5. Click Start to begin the calculation.

6. The calculation should take only a few seconds. Once complete the bivariate and univariate statistics are displayed in the Results section, including the R^2 statistic.

2.4.7 Making Predictions for Additional Molecules

Once an LIA model predicting binding affinities has been created and validated, binding affinities may be predicted for any molecule for which the three LIA terms have been generated. The steps are the same as was done for the test set above.

1. Run a Liaison calculation to generate the three LIA descriptors for your own ligands.
2. Import the molecules into the Project Table and ensure that they are selected.

   You can import them using either the Liaison panel or the Import panel.
3. Open the Strike Predict Based on QSAR Model panel.
4. Ensure that the LIA model is selected in the Results table.
5. Click Start.

   The generated predictions are added automatically to the Project Table as Predicted ActivityX.1 where X is the original model number for the LIA model.

2.5 Generating and Applying LiaScore and ELR Binding Affinity Models

In addition to the LIA model, other models of binding affinity may be generated using Liaison calculated properties. From the LiaScore a one-descriptor model may be created. The procedure is the same as was illustrated for the LIA model, though instead of selecting the three Liaison <Uvdw>, Liaison <Uele>, and Liaison <Ucav> terms from the Select descriptors to be included in the model table of the Build QSAR Model panel, simply select the LiaScore descriptor. Model generation, validation, and application then follow exactly as with the LIA model. Sample Strike input and output for the generation and application of the LiaScore only model for the current dataset may be found in:

```
$SCHRODINGER/impact-v version/tutorial/liaison/analysis/ls_*
```

For those interested in expanding the range of descriptors to correlate with binding affinities, an extended linear response (ELR) model may be most appropriate. An example of an ELR model would be to use the LiaScore components, eight descriptors named with a lia prefix, to generate a binding affinity model. With an ELR model you can take advantage of Strike's automatic variable selection or sophisticated partial least squares and principal component linear

fitting. Sample input and output for the generation and application of an ELR model using automatic variable selection starting with the eight LiaScore components may be found in:

```
$SCHRODINGER/impact-v version/tutorial/liaison/analysis/liacomp_*
```

Protein and Ligand Preparation

The quality of Liaison results depends on reasonable starting structures for both the protein and the ligand. Schrödinger offers a comprehensive protein preparation facility in the Protein Preparation Wizard, which is designed to ensure chemical correctness and to optimize protein structures for use with Glide and other products. Likewise, Schrödinger offers a comprehensive ligand preparation facility in LigPrep. It is strongly recommended that you process protein and ligand structures with these facilities in order to achieve the best results.

3.1 Protein Preparation

A typical PDB structure file consists only of heavy atoms, can contain waters, cofactors, and metal ions, and can also be multimeric. The structure generally has no information on bonding or charges. Terminal amide groups can also be misaligned, because the X-ray structure analysis cannot usually distinguish between O and NH_2. For Liaison calculations, which use an all-atom force field, atom types and bond orders must be assigned, the charge and protonation states must be corrected, side chains reoriented if necessary, and steric clashes relieved.

This section provides an overview of the protein preparation process. The entire procedure can be performed in the Protein Preparation Wizard panel, which you open from the Workflows menu on the main toolbar. This tool and its use is described in detail in Chapter 2 of the *Protein Preparation Guide*.

After processing, you will have files containing refined, hydrogenated structures of the ligand and the ligand-receptor complex. The prepared structures are suitable for use with Liaison. In most cases, not all of the steps outlined need to be performed. See the descriptions of each step to determine whether it is required.

You may on occasion want to perform some of these steps manually. Detailed procedures are described in Chapter 3 of the *Protein Preparation Guide*.

1. Import a ligand/protein cocrystallized structure, typically from PDB, into Maestro.

   The preparation component of the protein preparation facility requires an identified ligand.

2. Simplify multimeric complexes.

   For computational efficiency it is desirable to keep the number of atoms in the complex structure to a minimum. If the binding interaction of interest takes place within a single

subunit, you should retain only one ligand-receptor subunit to prepare for Liaison. If two identical chains are both required to form the active site, neither should be deleted.

- Determine whether the protein-ligand complex is a dimer or other multimer containing duplicate binding sites and duplicate chains that are redundant.
- If the structure is a multimer with duplicate binding sites, remove redundant binding sites and the associated chains by picking and deleting molecules or chains.

3. Locate any waters you want to keep, then delete all others.

   Water molecules in the crystallographic complex are generally not used unless they are judged critical to the functioning of the protein–ligand interaction. When waters are used, they are later included in the protein as "structural" waters. Keeping structural waters is more likely to be important for Liaison than for other programs such as Glide, where making a site more accessible by removing all waters may be necessary for docking.

   These waters are identified by the oxygen atom, and usually do not have hydrogens attached. Generally, all waters (except those coordinated to metals) are deleted, but waters that bridge between the ligand and the protein are sometimes retained. If waters are kept, hydrogens will be added to them by the preparation component of the protein preparation job. Afterwards, check that these water molecules are correctly oriented.

4. Adjust the protein, metal ions, and cofactors.

   Problems in the PDB protein structure may need to be repaired before it can be used. Incomplete residues are the most common errors, but may be relatively harmless if they are distant from the active site. Structures that are missing residues near the active site should be repaired.

   Metal ions in the protein complex cannot have covalent bonds to protein atoms. The MacroModel atom types for metal ions are sometimes incorrectly translated into dummy atom types (Du, Z0, or 00) when metal-protein bonds are specified in the input structure. Furthermore, isolated metal ions may erroneously be assigned general atom types (GA, GB, GC, etc.).

   It may be necessary to adjust the protonation of the protein, which is crucial when the receptor site is a metalloprotein such as thermolysin or an MMP. In such a case, Glide assigns a special stability to ligands in which anions coordinate to the metal center. To benefit from this assignment, groups such as carboxylates, hydroxamates, and thiolates must be anionic. The protein residues that line the approach to the metal center (such as Glu 143 and His 231 in thermolysin) need to be protonated in a manner compatible with the coordination of an anionic ligand such as a carboxylate or hydroxamate. The co-crystallized complex therefore needs to be examined to determine how the protein and the ligands should be protonated. In some cases, two or more protonation states of the protein

may need to be used in independent docking experiments to cover the range of physically reasonable ligand dockings.

Cofactors are included as part of the protein, but because they are not standard residues it is sometimes necessary to use Maestro's structure-editing capabilities to ensure that multiple bonds and formal charges are assigned correctly.

- Fix any serious errors in the protein.
- Check the protein structure for metal ions and cofactors.
- If there are bonds to metal ions, delete the bonds, then adjust the formal charges of the atoms that were attached to the metal as well as the metal itself.
- Set charges and correct atom types for any metal atoms, as needed.
- Set bond orders and formal charges for any cofactors, as needed.

5. Adjust the ligand bond orders and formal charges.

   If the complex structure contains bonds from the ligand or a cofactor to a protein metal, they must be deleted. Glide models such interactions as van der Waals plus electrostatic interactions. Glide cannot handle normal covalent bonds to the ligand, such as might be found in an acyl enzyme.

   If you are working with a dimeric or large protein and two ligands exist in two active sites, the bond orders have to be corrected in both ligand structures.

6. Run a restrained minimization of the protein structure.

   This is done with `impref`, and should reorient side-chain hydroxyl groups and alleviate potential steric clashes.

7. Review the prepared structures.

   - If problems arise during the restrained minimization, review the log file, correct the problems, and rerun.
   - Examine the refined ligand/protein/water structure for correct formal charges, bond orders, and protonation states and make final adjustments as needed.

3.2 Checking the Protein Structures

After you have completed the protein preparation, you should check the completed ligand and protein structures.

3.2.1 Checking the Orientation of Water Molecules

You only need to perform this step if you kept some structural waters. Reorienting the hydrogens is not strictly necessary, as their orientation should have been changed during refinement, but it is useful to check that the orientation is correct.

If the orientation is incorrect, reorient the molecules by using the procedure outlined in Section 3.9 of the *Protein Preparation Guide*.

When you have corrected the orientation of the retained water molecules, you should run a refinement on the adjusted protein-ligand complex.

3.2.2 Checking for Steric Clashes

You should make sure that the prepared site accommodates the co-crystallized ligand in the restraint-optimized geometry obtained from the structure preparation.

Steric clashes can be detected by displaying the ligand and protein in Maestro and using the Contacts folder in the Measurements panel to visualize bad or ugly contacts. Maestro defines bad contacts purely on the basis of the ratio of the interatomic distance to the sum of the van der Waals radii it assigns. As a result, normal hydrogen bonds are classified as bad or ugly contacts. By default, Maestro filters out contacts that are identified as hydrogen bonds, and displays only the genuine bad or ugly contacts.

If steric clashes are found, repeat the restrained optimization portion of the protein preparation procedure, but allow a greater rms deviation from the starting heavy-atom coordinates than the default of 0.3 Å. Alternatively, you can apply an additional series of restrained optimizations to the prepared ligand-protein complex to allow the site to relax from its current geometry.

3.2.3 Resolving H-Bonding Conflicts

You should look for inconsistencies in hydrogen bonding to see whether a misprotonation of the ligand or the protein might have left two acceptor atoms close to one another without an intervening hydrogen bond. One or more residues may need to be modified to resolve such an acceptor-acceptor or donor-donor clash.

Some of these clashes are recognized by the preparation process but cannot be resolved by it. The preparation process may have no control over other clashes. An example of the latter typically occurs in an aspartyl protease such as HIV, where both active-site aspartates are close to one or more atoms of a properly docked ligand. Because these contact distances fall within any reasonable cavity radius, the carboxylates are not subject to being neutralized and will both be represented as negatively charged by the preparation process. However, when the ligand inter-

acts with the aspartates via a hydroxyl group or similar neutral functionality, one of the aspartates is typically modeled as neutral.

If residues need to be modified, follow these steps:

1. Place the refined protein-ligand complex in the Workspace.
2. Examine the interaction between the ligand and the protein (and/or the cofactor).
3. Use your judgment and chemical intuition to determine which protonation state and tautomeric form the residues in question should have.
4. Use the structure-editing capabilities in Maestro to resolve the conflict (see Section 3.8 of the *Protein Preparation Guide* for procedures).
5. Re-minimize the structure.

It is usually sufficient to add the proton and perform about 50 steps of steepest-descent minimization to correct the nearby bond lengths and angles. Because this optimizer does not make large-scale changes, the partial minimization can be done even on the isolated ligand or protein without danger of altering the conformation significantly. However, if comparison to the original complex shows that the electrostatic mismatch due to the misprotonation has appreciably changed the positions of the ligand or protein atoms during the protein-preparation procedure, it is best to reprotonate the original structure and redo the restrained minimization.

3.3 Ligand Preparation

To give the best results, the structures that are used must be good representations of the actual ligand structures as they would appear in a protein-ligand complex. This means that the structures supplied to Liaison must meet the following conditions:

1. They must be three-dimensional (3D).
2. They must have realistic bond lengths and bond angles.
3. They must each consist of a single molecule that has no covalent bonds to the receptor, with no accompanying fragments, such as counter ions and solvent molecules.
4. They must have all their hydrogens (filled valences).
5. They must have an appropriate protonation state for physiological pH values (around 7).

   For example, carboxylic acids should be deprotonated and aliphatic amines should be protonated. Otherwise a neutral aliphatic amine could improperly act as a hydrogen-bond acceptor in the docking calculations, or could occupy a hydrophobic region without

incurring the large desolvation penalty that XP Glide docking would have assessed if the amine had been properly protonated.

Protonation states are particularly crucial when the receptor site is a metalloprotein such as thermolysin or a MMP. If the metal center and its directly coordinated protein residues have a net charge, Glide assigns a special stability to ligands in which anions coordinate to the metal center. To benefit from this assignment, groups such as carboxylates, hydroxamates, and thiolates must be anionic. If there is no net charge, Glide gives no preference to anions over neutral functional groups.

6. They must be supplied in Maestro, SD, Mol2, or PDB format.

   Maestro transparently converts SD, MacroModel, Mol2, PDB, and other formats to Maestro format during structure import. However, Glide has no direct support for other formats, so you should ensure that your structures are in Maestro, SD, Mol2, or PDB format before starting Liaison jobs.

All of the above conditions can be met by using LigPrep to prepare the structures. Use of LigPrep is described in the next section.

3.3.1 Using LigPrep for Ligand Preparation

The Schrödinger ligand preparation product LigPrep is designed to prepare high quality, all-atom 3D structures for large numbers of drug-like molecules, starting with 2D or 3D structures in SD, Maestro, or SMILES format. LigPrep can be run from Maestro or from the command line. For detailed information on LigPrep, see the *LigPrep User Manual*.

To run LigPrep, you must have a LigPrep license. The MacroModel commands `premin` and `bmin` require LigPrep licenses when run in a LigPrep context, and are limited to a restricted set of commands when run using a LigPrep license. LigPrep can be run from Maestro or from the command line.

The LigPrep process consists of a series of steps that perform conversions, apply corrections to the structures, generate variations on the structures, eliminate unwanted structures, and optimize the structures. Many of the steps are optional, and are controlled by selecting options in the LigPrep panel or specifying command-line options. The steps are listed below.

1. Convert structure format.
2. Select structures.
3. Add hydrogen atoms.
4. Remove unwanted molecules.
5. Neutralize charged groups.

6. Generate ionization states.
7. Generate tautomers.
8. Filter structures.
9. Generate alternative chiralities.
10. Generate low-energy ring conformations.
11. Remove problematic structures.
12. Optimize the geometries.
13. Convert output file.

The LigPrep panel allows you to set up LigPrep jobs in Maestro. Choose LigPrep from the Applications menu to open the panel. For details of panel options and operation, see Chapter 2 of the *LigPrep User Manual*.

The simplest use of LigPrep produces a single low-energy 3D structure with correct chiralities for each successfully processed input structure. LigPrep can also produce a number of structures from each input structure with various ionization states, tautomers, stereochemistries, and ring conformations, and eliminate molecules using various criteria including molecular weight or specified numbers and types of functional groups present.

The default options in the LigPrep panel remove unwanted molecules, add hydrogens, and minimize the ligand structure (performing a 2D-3D conversion, if necessary). Below are notes on panel options that produce more than one output structure per input structure.

The `stereoizer` can generate two stereoisomers per chiral center in the ligand, up to a specified maximum. There are three Stereoisomers options.

> The first two options, Retain specified chiralities (the default) and Determine chiralities from 3D structure, generate both isomers only at chiral centers where chirality is unspecified or indeterminate; centers with known chirality retain that chirality. The difference is that Retain specified chiralities takes its chirality data from the input file (SD or Maestro), while Determine chiralities from 3D structure ignores input file chiralities and takes chirality information from the 3D geometry.
>
> Generate all combinations varies the stereochemistry up to a maximum number of structures specified by Generate at most *max* per ligand. The default maximum is 32.

The Ionization options allow you to generate all the ligand protonation states that would be found in the specified pH range. The Ionization options are:

> Retain original state

Neutralize

Generate possible states at target pH *target* +/- *range*. This is the default, and can generate several different output structures for each input structure. The default pH *target* is 7.0 with a +/- *range* of 2.0, so the default pH range is 5.0 – 9.0. Both the target and range settings can be changed. You can use either the `ionizer` or Epik to generate ionization states. Epik is a separate product, so you must purchase this product to use it.

Generate low energy ring conformations: *number* per ligand. The default is to generate only the lowest energy conformation.

Desalt is selected by default.

Generate tautomers is selected by default. The `tautomerizer` generates up to 8 tautomers per ligand, selecting the most likely tautomers if more than 8 are possible. If you are sure that the input structures are already in the correct tautomeric form for docking to a particular target, then the `tautomerizer` should be turned off by deselecting Generate tautomers.

3.3.2 Using Other Programs for Ligand Preparation

If you prefer to prepare the ligands with other programs, you can do so. Schrödinger software installations include a number of utilities that can be used to perform some of the above tasks. These utilities are also used by LigPrep. One of these, the Ionizer, can be used to prepare ligands in the required protonation states. Some of the other tasks can be performed as follows:

- Hydrogen atoms can be added in Maestro with either the Add hydrogens toolbar button:

  or the Hydrogen Treatment panel (select Hydrogen Treatment from the Edit menu).

  Hydrogen atoms can also be added (or removed) using the utility `applyhtreat`, which is described in Appendix D of the *Maestro User Manual*.

- Structure file format conversion can be done from the command line with utilities such as `pdbconvert`, `sdconvert`, and `maemmod`—see Appendix D of the *Maestro User Manual*.

Running Liaison

Liaison is a method of predicting ligand-protein binding free energies using a model that has been fitted to known binding energy values. The process involves two steps, a fitting step and a predicting step. Each step is carried out as two tasks, a simulation task and an analysis task.

The Liaison panel is used to set up and run the simulation tasks. It runs the Ligand & Structure-Based Descriptors script (`lsbd`) to set up and run the Liaison job. The analysis tasks are performed using the Build QSAR Model panel of Strike. For more information on Strike, see the *Strike User Manual*. A tutorial introduction to the Liaison tasks is given in Chapter 2.

4.1 Overview of Liaison Tasks

Before you run a Liaison simulation, you should ensure that the receptor and the ligands are properly prepared, as described in Chapter 3. You should also ensure that the ligand structure file includes the known binding energies.

To run a Liaison simulation:

1. Specify the systems to be simulated in the Specify structures section of the Systems tab. You can take the receptor and ligands from a Glide pose viewer file, or you can take the receptor from the Workspace or a file and take the ligands from the Project Table or a file.
2. Specify the kind of system to be simulated in the Job options section of the Parameters tab.
3. Set constraints if required in the Specify restrained/frozen shells section of the Parameters tab.
4. Click Start.
5. In the Start dialog box, set the job name and select the host and number of processors.

   If you want to choose a remote machine or batch queue as the host for the job, ensure that the current working directory is mounted on the remote host. Liaison input files are written to the current working directory. Liaison does not have the ability to copy input files from a local directory to a remote scratch directory.

When the job finishes, the results are incorporated into the Project Table. The scores and the various components that are used in the analysis task are added as properties. If the job does not incorporate, select it in the Monitor panel and click Monitor.

To run a Liaison fitting job:

1. Select the results of a Liaison simulation:

   - If the simulation results are already in the Project Table, select the relevant entries, and choose Build QSAR Model from the Strike submenu of the Applications menu.

     If the simulation results include both the training set and the test set, you should select the training set for the fitting.

   - If the simulation results are not incorporated, click Browse in the Analysis tab of the Liaison panel, navigate to and select the file *jobname*`-final.mae`, and click Fit or Predict in Strike.

   The Build QSAR Model panel of Strike opens, with the Liaison results loaded.

2. Select the descriptors required for the fit from the list (click, shift-click, control-click):

   - For an LRM model, select Uvdw, Uele, and Ucav.
   - For a LiaScore model, select LiaScore.

3. Choose Multiple Linear Regression from the Regression method option menu.

4. For Descriptors, ensure that Use all selected is selected.

5. Click Choose to choose the activity property.

   If the activity property is not in the Project Table, you can import it from a CSV file, but you must do this before clicking Choose.

6. Click Start.

7. Set job options in the Start dialog box, and click Start.

To run a Liaison prediction job:

1. Select the results of a Liaison simulation for the test set (the ligands whose binding energy you want to predict) in the Project Table.

2. Choose Predict from the Strike submenu of the Applications menu.

3. Select the model you want to apply from the list.

4. Click Start.

5. Set job options in the Start dialog box, and click Start.

When the job finishes, you can view the results by clicking View. The predicted values are also added to the Project Table.

4.2 The Liaison Panel

The main part of the Liaison panel consists of three tabs:

- Systems tab
- Parameters tab
- Analysis tab

Below these tabs on the left are the Start button, for starting the Liaison simulation, and the Write button, for writing the Liaison input files for later use from the command line.

To open the Liaison panel, choose Liaison from the Applications menu in the main window.

4.2.1 Systems Tab

This tab has a single section, Specify structures, for defining the system to be studied.

Take complexes from a Maestro pose viewer file option and controls

Use a Maestro (Glide) pose viewer file as the source of the receptor and the ligands. Click Browse to navigate to the file, or enter the path in the File text box.

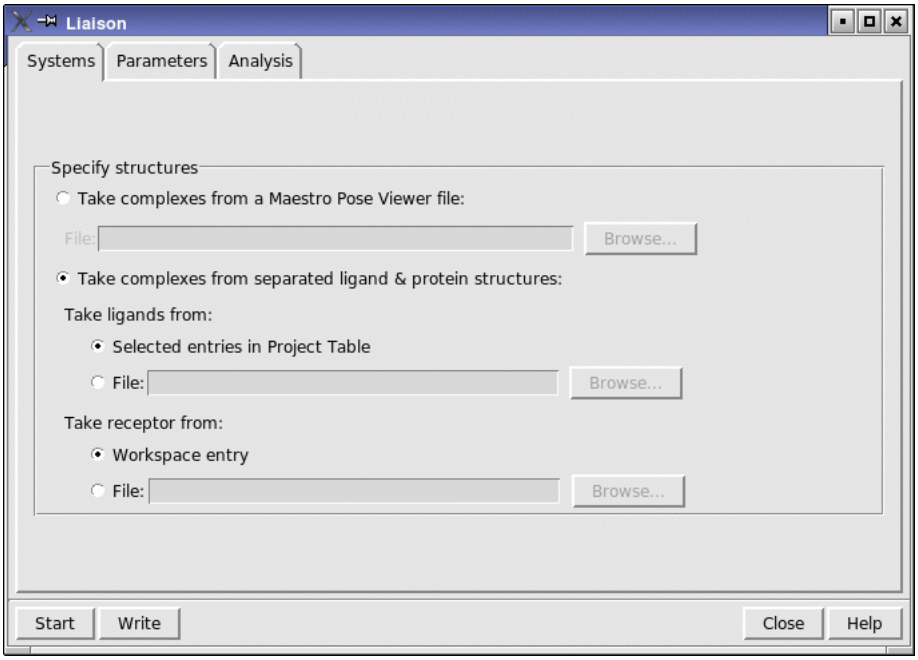


Figure 4.1. The Systems tab of the Liaison panel.

Take complexes from separated ligand & protein structures option and controls

Specify the receptor and the ligand structures separately.

Take ligands from: Select either Selected entries in Project Table or File. If you select a file, click Browse to navigate to the file, or enter the path in the File text box.

Take receptor from: Select either Workspace entry or File. If you select a file, click Browse to navigate to the file, or enter the path in the File text box.

4.2.2 Parameters Tab

This tab has two sections, one for setting the simulation parameters (Job options) and one for defining restrained and frozen shells (Specify restrained/frozen shells).

The options that apply to both the complex and the free ligand are in the upper part of the Job options section. Options that apply to one or the other are in tabs labeled Ligand Simulation and Complex Simulation in the lower part of this section. The controls in these tabs are identical, and are described below.

Sampling method option menu

Choose the method for performing the simulation, from Minimization, Hybrid Monte Carlo, or Molecular Dynamics.

Minimization algorithm option menu

Choose an algorithm for performing the minimization steps in any of the three sampling methods. Available algorithms are Truncated Newton, Conjugate Gradient, and Steepest Descent.

Simulation temperature text box

Specify the temperature of the simulation in K. Default: 300 K Not available with the Minimization sampling method.

Temperature relaxation time text box

Specify the time scale, in picoseconds, on which heat is exchanged with the heat bath. Default: 10 ps Not available with the Minimization sampling method.

Residue-based cutoff distance text box

Set the value (in Å) for the cutoff distance of non-bonded interactions. All pairwise interactions of an atom in one residue with an atom in another residue are included on the non-bonded pair list if any such pair of atoms is separated by this distance or less. Default: 15 Å.

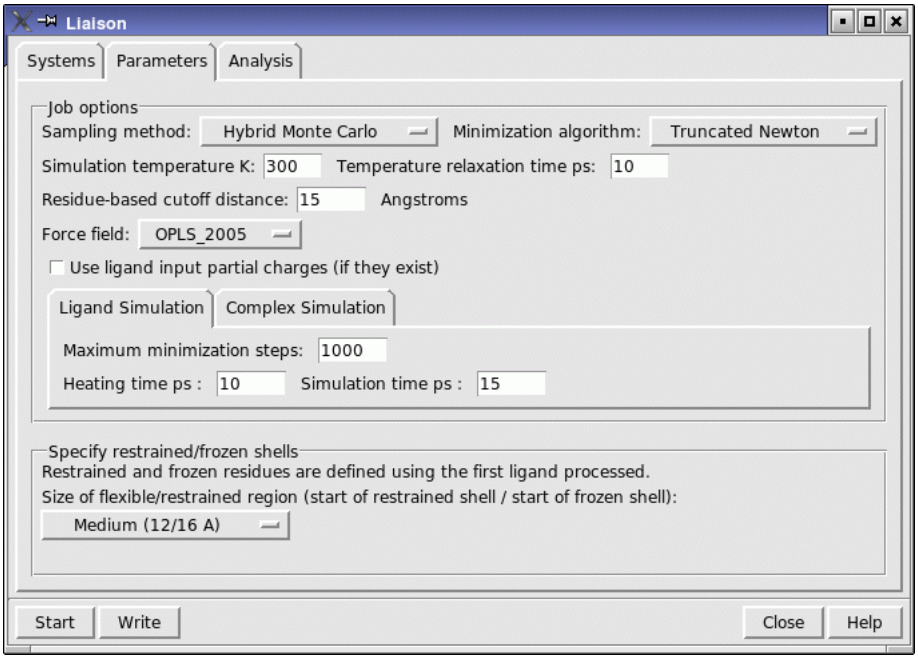


Figure 4.2. The Parameters tab of the Liaison panel.

Force field option menu

Select the force field. The force field options are OPLS_2005 (the default) and OPLS_2001. Liaison simulations are run using Surface Generalized Born (SGB) continuum solvation. Select the OPLS_2005 force field if you want the calculation to use the improved parametrized nonpolar model instead of the default SGB terms that are used with OPLS_2001.

Use ligand input partial charges (if they exist) option

Select this option to use the input charges for the ligand in Liaison calculations (for both the free and the bound state).

Maximum minimization steps text box

Specify the maximum steps to take during any minimization. Can be set independently for ligands and complexes. Default: 1000.

Heating time text box

Set the time in picoseconds over which the system is heated before the LIA task is started, in an HMC or MD simulation. Default: 10 ps.

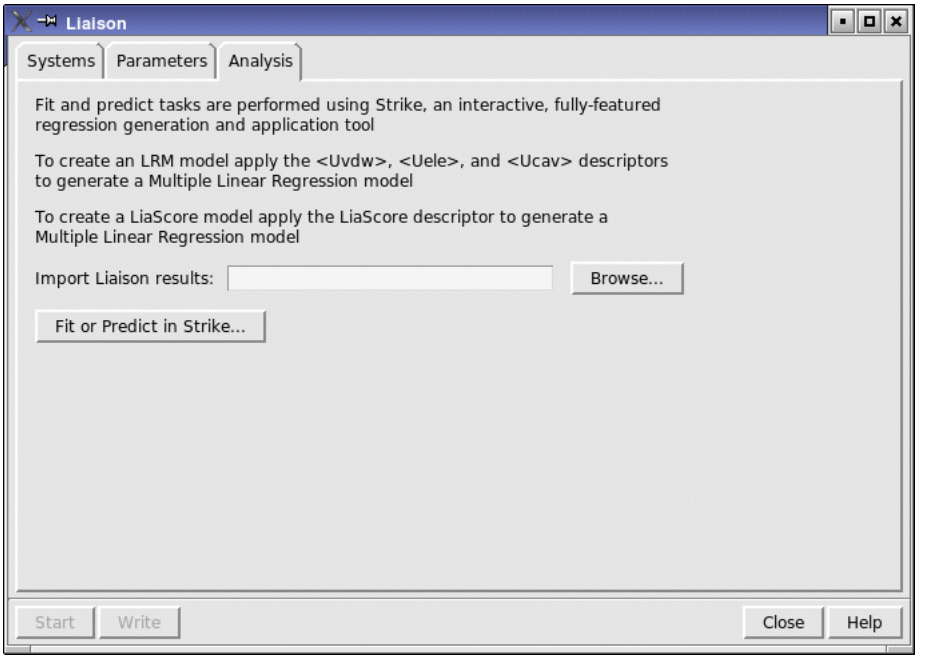


Figure 4.3. The Analysis tab of the Liaison panel.

Simulation time text box

Set the simulation time for the LIA task, in an HMC or MD simulation. In this task the averages for the van der Waals, Coulombic, reaction field and cavity terms are determined.

In the Specify restrained/frozen shells section, you can specify which residues are to be restrained or frozen, by choosing cutoffs for the start of the restrained region and the start of the frozen region. Residues with any atoms inside the restrained region boundary are treated as flexible. Residues with any atoms inside the frozen region boundary but outside the restrained region boundary are restrained. Residues with all atoms outside the frozen region boundary are frozen. You can choose from a range of options for the location of the two boundaries.

Note: You should make sure that you make the same choice of restrained and frozen shells when you run Liaison for a particular system, otherwise the results will be invalid.

4.2.3 Analysis Tab

This tab provides an interface to Strike for performing the fit and predict steps of the Liaison analysis. You can use Strike independently. If you use these controls, you must specify the file that contains the results of the Liaison simulation, named jobname-final.mae. The structures in this file are imported into the Project Table; when you click Fit or Predict in Strike, these entries

are selected in the Project Table and used as input to Strike. Clicking this button opens the Strike Build QSAR Model panel. For more information, see Chapter 3 of the *Strike User Manual*.

4.3 Running Liaison From the Command Line

Once you have set up a Liaison job in Maestro, you can click Write to write out the input file, then run the Liaison simulation job from the command line with the following command:

`$SCHRODINGER/lsbd [`*job-options*`]` *input-file*

where *job-options* are the standard Job Control options, described in the *Job Control Guide*.

Technical Notes for Liaison

5.1 Models of Ligand Binding

The typical binding site for a ligand is the active-site cavity of a protein receptor. When no ligand is present, this cavity is filled with water molecules. When a ligand binds to the protein in this cavity, it displaces water molecules in the active site, which return to bulk solvent.

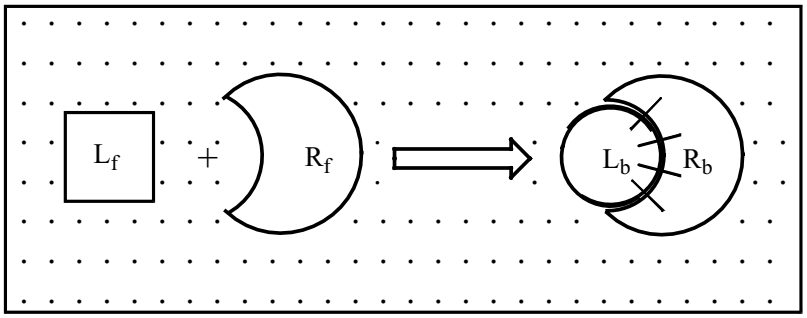


Figure 5.1. Schematic view of ligand binding in a receptor cavity with displacement of water. The ligand and the receptor both change, to a greater or lesser extent, from a "free" (f) to a "bound" (b) conformation.

The principal factors determining the strength and specificity of binding are as follows:

- The degree to which hydrophobic groups on the ligand interact with hydrophobic pockets or patches on the protein surface to release water into the bulk. This release is favorable both energetically (more hydrogen bonds are formed by the released water molecules) and entropically (the released waters are less constrained orientationally and are no longer confined to a restricted cavity).
- The extent to which the ligand forms hydrogen bonds or metal ligations in hydrophilic regions with appropriately placed polar or charged groups on the receptor. Such complementarity is essential for achieving adequate binding affinity and specificity.
- The ease with which the ligand fits into the protein cavity. An important question is what it costs the ligand (and the protein) in energy and/or entropy to accomplish this fit—i.e., to change from the free to the bound conformation, as indicated schematically in Figure 5.1. Energy will be required if the ligand has to be distorted away from its natu-

rally preferred, low-energy conformation into a higher-energy conformation when it binds to the receptor. At the same time, entropy will be lost if the ligand is very flexible in solution and then is confined to a small number of conformations in the receptor cavity. Both effects, as well as similar restrictions on the conformation of protein side chains, act against binding.

5.1.1 Limitations of Free-Energy Perturbation

Molecular simulation methods have been used to calculate binding free energies of protein-ligand calculations since the pioneering applications of free energy perturbation (FEP) approaches by McCammon, Kollman, Jorgensen, and others approximately 20 years ago. During the past two decades, many FEP calculations have been carried out in academic groups and in pharmaceutical and biotechnology companies. But while notable successes have been achieved, FEP methods are in limited use in drug-discovery projects, for several reasons:

- FEP calculations are typically limited to small changes in ligand structure, restricting the applicability to the very last phase of lead optimization.
- FEP calculations are very expensive computationally, and often cannot be completed on a time scale compatible with the schedule of a given drug-discovery project.
- Inaccuracies in force fields and sampling methods can lead to errors in FEP predictions.

5.1.2 Advantages of Linear Response Methods

The limitations of FEP motivated the development of linear-response (LR) methods by Aqvist several years ago (Hansson, T.; Aqvist, *J. Protein Eng.* **1995,** *8,* 1137–1145). Since that time, studies by Jorgensen and others have shown that LR methods can effectively address the above difficulties. In comparison with the FEP approach, the advantages of LRM are as follows:

- In contrast to FEP, where a large number of intermediate “windows” must be evaluated, LRM requires simulations only of the ligand in solution and the ligand bound to the protein. The idea is that one views the binding event as replacement of the aqueous environment of the ligand with a mixed aqueous/protein environment.
- Only interactions between the ligand and either the protein or the aqueous environment enter into the quantities that are accumulated during the simulation. The protein-protein and protein-water interaction are part of the “reference” Hamiltonian, and hence are used to generate conformations in the simulation, but are not used as descriptors in the resultant model for the binding free energy. This eliminates a considerable amount of noise in the calculations—for example, that arising from variations in the total energy that result because slightly different geometries of the protein are obtained for each ligand molecule simulated.

- As long as the binding modes of the ligand are fundamentally similar, LRM calculations can be applied to ligands that differ significantly in chemical structure.
- LRM calculations are less computationally expensive than FEP calculations.
- The LRM approach allows the binding-energy model to be calibrated by using a training set of compounds for which experimental binding affinities are known. The use of the energy terms as descriptors in the fitting equation introduces an empirical element that allows some of the limitations in the theoretical framework (for example, the neglect of the cost in energy and entropy of fitting the ligand into the protein site) and the physical representation (as reflected by errors in the force field or solvation model) to be partially absorbed into the parameterization. Moreover, some of the steps involved in the binding event, such as the removal of water from the protein cavity and subsequent introduction of the ligand, are not inherently linear. If the linear-response approximation was rigorously valid, the coefficients of the terms would each be 0.5, corresponding to the mean-value approximation to the "charging" integral. In practice, optimization of the fitting parameters yields coefficients that are significantly different from the ideal value of 0.5. This empirical element sacrifices generality: the method probably requires the ligands to have similar binding modes, and new parameters must be developed for each receptor. In return, one can obtain a reasonable level of accuracy with a modest expenditure of CPU time, under assumptions that are quite reasonable for many structure-based drug-design projects.

5.1.3 Advantages of Liaison

Liaison—Schrödinger's continuum-solvent implementation of LRM—has a number of highly attractive features in addition to those listed in Section 5.1.2:

- The use of the Surface Generalized Born (SGB) continuum model greatly speeds the calculation because the various interaction terms converge much faster than in an explicit-solvent simulation. As a result, the required CPU time is reduced by a factor of 10 or more.
- An even greater reduction in computational effort can be achieved by using a simple energy minimization protocol, rather than a molecular dynamics or Hybrid Monte Carlo simulation, and obtaining the LRM fitting data from the lowest-energy point reached by the minimization. While there is sometimes a small degradation of accuracy as compared to a simulation, the speed of the calculation is qualitatively enhanced.
- When "energy-minimization" sampling is coupled with a highly efficient Truncated Newton minimizer, Liaison calculations are fast enough to be applied routinely in a contemporary drug-discovery context. This approach makes Liaison very attractive for screening a large number of ligands.

- Schrödinger's automatic atom-typing scheme for the OPLS-AA force field (Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. *J. Am. Chem. Soc.* **1996,** *118,* 11223–11235) assigns charges, van der Waals, and valence parameters with no human intervention. A key feature of OPLS-AA is that, via fitting to liquid-state simulations, excellent reproduction of condensed-phase properties is obtained.

The Maestro graphical user interface makes it easy to set up and run Liaison calculations. First, a training set of compounds for which experimental binding affinities are available can be used to generate simulation data that are then employed to determine the optimal LRM parameters. Once the parameters have been determined, libraries of compounds with unknown binding affinities can be run and their binding affinities can be predicted. Alternatively, Liaison can perform scoring on relaxed protein-ligand complexes, allowing direct prediction of ligand binding energies.

It is important that the protein structures be correctly prepared. See the *Protein Preparation Guide*for a description of Schrödinger's protein-preparation facility.

5.2 LiaisonScore Binding Free Energy Model

Liaison calculates a scoring function, similar to GlideScore 3.5 SP, over the course of the LRM simulation. This function, previously known as "GlideScore in Liaison" is now called LiaisonScore. The average LiaisonScore found in the simulation can be used to predict binding energies using the formula

$$\Delta G = a(\langle LiaisonScore \rangle) + b$$

instead of the LRM equation described above. The LiaisonScore binding energy model can be selected in the Analysis tab. The Analyze task can then be used to derive values for the a and b fitting coefficients.

This capability allows Liaison to be employed in the early stages of a drug-discovery project, i.e., before a set of ligands with known binding affinities suitable for training a LRM model is available. This approach transcends scoring in Glide itself because it allows the protein site to relax and enables the ligand to undergo full (not just torsional and rigid-body) optimization.

Advantages of LiaisonScore

Recall that docking with Glide is normally done with the vdW radii of non-polar ligand and/or protein atoms scaled back by 10 to 20%. This rescaling creates more room in the rigid protein site and implicitly allows for "breathing motions" a protein site may carry out to accommodate a ligand that is slightly larger in some dimension than the ligand co-crystallized with the protein. But while some protein sites may readily expand (or contract) upon ligand binding, others may be less adaptable, and the device of rescaling radii used by Glide will miss this

distinction. Moreover, Glide further compensates for the limitations of docking into a rigid protein site by allowing intramolecular contacts within the docked ligand pose to be shorter than are physically realistic. These two aspects mean that Glide at times will give good scores to ligands that are too large to bind in a given receptor site. There can also be cases in which the rigid site is somewhat too large for an active binder, particularly with the scaling down of non-polar radii. As a result an active ligand may score more poorly than it should.

In principle, such false positives and false negatives can be eliminated by repeating the Glide scoring after relaxing the ligand-protein complex. In this way, if the protein can readily adapt to the true (not Glide compacted) size and shape of the docked ligand, the recomputed GlideScore value should be good. But if the protein site cannot respond appropriately, a poor re-scored GlideScore value should be obtained. Note, however, that sites that are much too small will simply generate incorrect docked poses, and minimization will not transform such a pose into a correctly docked structure.

Calibration of GlideScores

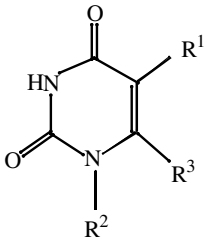
If a training set of ligands with known binding affinities is available, calibrated GlideScore values can be obtained as

$$\text{GlideScore(calibrated)}_i = a\ \text{GlideScore(raw)}_i + b \quad (1)$$

Calibration has no effect on the rank order of calculated binding affinities or on the computed correlation coefficient, but can place the computed GlideScores on the correct experimental scale and make it easier to interpret the calculated values.

5.3 Application to HIV Reverse Transcriptase

The Jorgensen group has been a leader in the development and testing of linear-response approaches for systems of pharmaceutical interest using explicit-solvent methods. An initial application to HIV reverse transcriptase examined the binding of 20 HEPT and 20 nevirapine analogs (*J. Med. Chem.* **2001,** *44,* 145–154). More recently, this work has been extended to encompass 200 different inhibitors covering 8 chemical classes (*J. Med. Chem.* **2002,** *45,* 2970–2987). Nevirapine is a FDA-approved anti-HIV drug of the non-nucleoside inhibitor class, and one of the HEPT analogs (MKC-442) has been in clinical trials. As shown in Figure 5.2, the variation in the R^2 sidechain is particularly sizable, ranging from hydrogen to methyl benzyl ether. It would be very difficult, and very time consuming, to examine structural variations of this magnitude using free-energy perturbation methods.



R^1 = Me, Et, i-Pr, c-Pr

R^3 = SPh, CH_2Ph, OPh, SPH-3,5-di-Me

R^2 = H, Me, Et, Pr, CH_2OCH_3, $CH_2OCH_2CH_3$, $CH_2OCH_2CH_2OH$, CH_2OCH_2Ph

Figure 5.2. Substituents for the HEPT series of inhibitors for HIV-RT.

The Liaison calculations presented in this section use input geometries for the ligands obtained from Glide 3.5 SP and XP dockings using Glide 3.5. Energy-minimization sampling was used in each case. Calibrated GlideScore values using Glide SP and Glide XP, LiaisonScores, and Liaison LRM-model predictions appear in the figures that follow.

Figure 5.3 shows the result obtained in the Glide 3.5 SP dockings, while Figure 5.4 shows the results obtained using Glide 3.5 XP scoring. As can be seen, SP Glide gives only a very rough correlation of the docked GlideScore values (shown as the calculated ΔG_{bind} values) with the experimental binding affinities. This is not surprising, as correlating experimental binding affinities will be beyond the capabilities of Glide docking and scoring in some cases, because of the limitations of rigid-receptor docking discussed above. Figure 5.4 shows that XP docking and scoring gives a considerably better correlation.

The results of applying the LiaisonScore binding energy model, using energy-minimization sampling and starting from the Glide 3.5 SP docked geometries for the ligands, are shown in Figure 5.5. The correlation is much tighter and the largest outliers are gone, showing that relaxation of the protein and docked ligand geometries has improved the results. These GlideScore values, unlike those in the previous two figures, have been put on the scale of the experimental binding affinities by applying the linear transformation shown in Equation (1). Thus, the predicted values now cover the approximately same range as the measured values. The slope of the fitted least-squares line is still less than 1, but is larger than those seen in Figure 5.3 and Figure 5.4.

The conventional LRM fit, again using energy-minimization sampling and starting from the Glide 3.5 SP ligand poses, is shown in Figure 5.6. As can be seen, LRM scoring produces a higher r^2 correlation coefficient and a lower RMS deviation between the calculated and observed binding affinities.

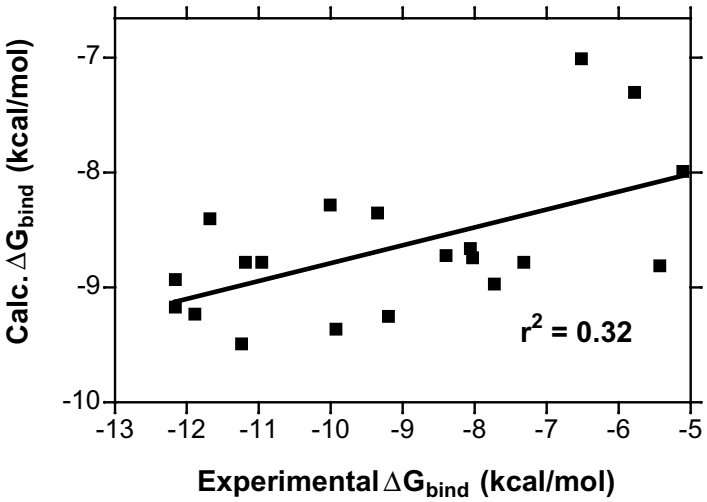


Figure 5.3. Correlation of calculated (GlideScore) and observed binding affinities for Glide 3.5 SP docking of HEPT inhibitors into HIV reverse transcriptase.

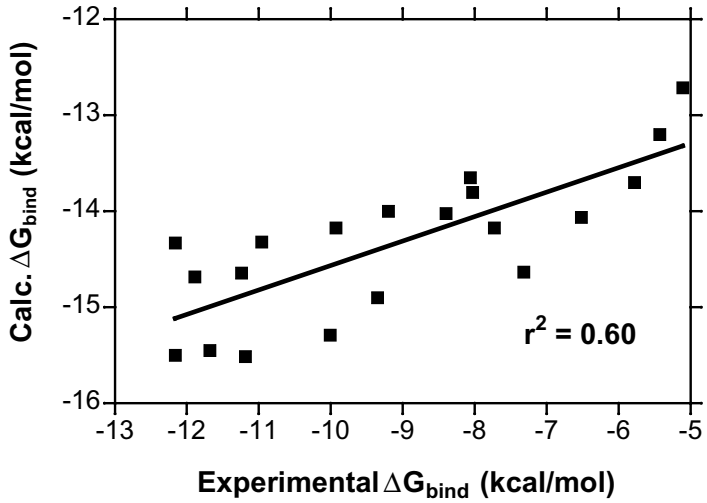


Figure 5.4. Correlation of calculated (GlideScore) and observed binding affinities for Glide 3.5 XP docking of HEPT inhibitors into HIV reverse transcriptase.

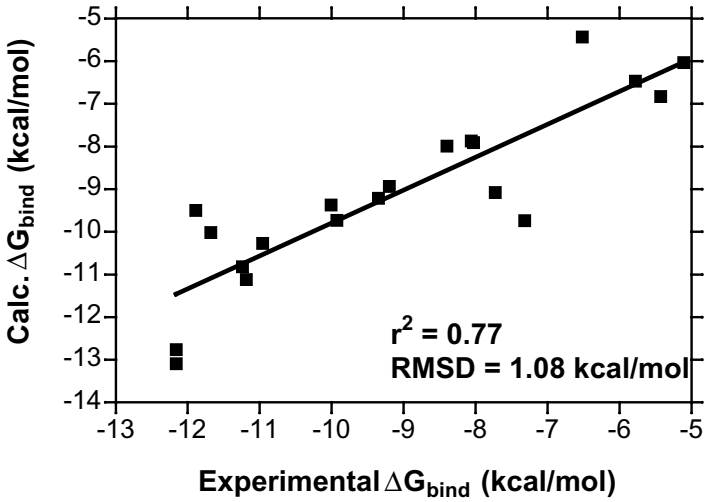


Figure 5.5. Correlation of LiaisonScores for the HEPT inhibitors for HIV reverse transcriptase, calculated by energy-minimization sampling starting from Glide 3.5 SP docked geometries.

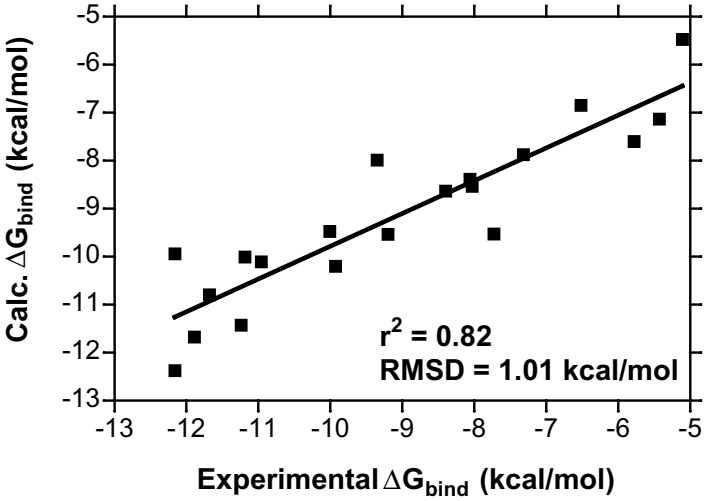


Figure 5.6. Correlation of Liaison LRM scoring for the HEPT series of inhibitors for HIV reverse transcriptase, calculated by energy-minimization sampling starting from Glide 3.5 SP docked geometries.

These results suggest that some of the limitations expected for rigid-receptor docking can be addressed by relaxing the docked complexes before scoring them with GlideScore or via a standard LRM fitting model. LRM scoring gives the best results in this instance, but it remains to be seen whether this will always be the case.

Normally we would expect even better results from energy-minimization sampling followed by rescoring in Liaison when starting from the XP-docked poses. This indeed is our recommended procedure. In this instance, however, the results obtained using the same two models gives correlations that are better than those shown in Figure 5.3 but are not as good as those in Figure 5.4 and Figure 5.5. This is counterintuitive, especially in view of the fact that XP docks only one ligand in a manner that is inconsistent with the other cases, and here the variation merely consists of a reversal of the positions adopted by the pseudo-symmetric C-R1 and N-R2groups (see Figure 5.2). In contrast, SP Glide docks three ligands in an inconsistent, and presumably incorrect, manner.

5.4 Application to ß-Secretase (BACE) Inhibitors

Tounge and Reynolds (*J. Med. Chem.* **2003,** *46,* 2074–2082) recently reported the application of Liaison to ten β-secretase inhibitors of the general structure shown in Figure 5.7, where R_1 is typically a BOC-capped dipeptide, and R_2 and R_3 are usually methyl or isopropyl. These neutral inhibitors have relatively high molecular weights and resemble HIV protease ligands in structure. Tounge and Reynolds also studied two nanomolar co-crystallized octapeptides that have two or three aspartate or glutamate residues and hence are negatively charged at physiological pH.

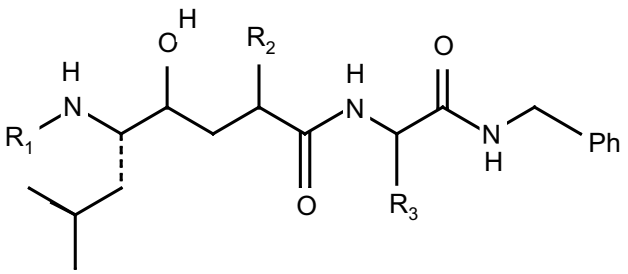


Figure 5.7. Schematic structure of neutral BACE inhibitors studied by Tounge and Reynolds.

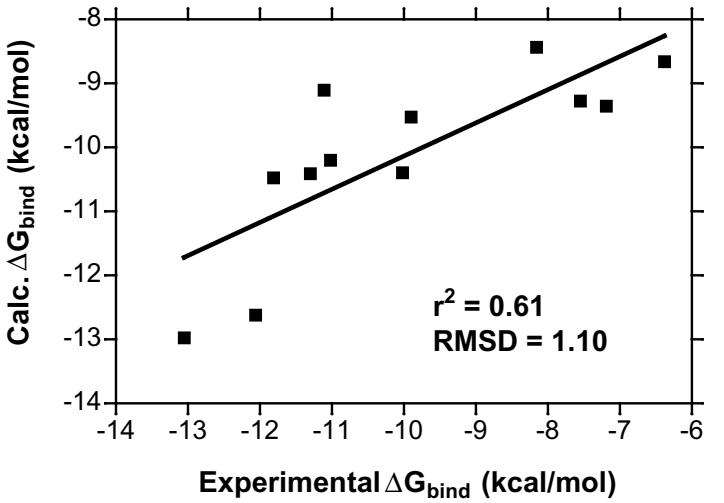


Figure 5.8. Correlation of predicted and observed binding affinities for BACE inhibitors using LRM scoring and energy-minimization sampling.

A plot of their data for energy-minimization sampling, taken from Figure 5 in their paper, is presented in Figure 5.8; hybrid Monte Carlo sampling gave a similar result. The correlation is reasonably good, but the negatively charged octapeptides (the two data points with the most negative calculated and observed binding affinities) fall off the line. This is not surprising, as it is notoriously difficult in binding energy correlations to get ligands having different charge states to fall on a common line.

Tounge and Reynolds also presented results for other simulation conditions that gave poorer fits to the experimental data. We nevertheless regard their results as promising, and intend to employ their series of BACE inhibitors in future efforts to improve Liaison.

Getting Help

Schrödinger software is distributed with documentation in PDF format. If the documentation is not installed in `$SCHRODINGER/docs` on a computer that you have access to, you should install it or ask your system administrator to install it.

For help installing and setting up licenses for Schrödinger software and installing documentation, see the *Installation Guide*. For information on running jobs, see the *Job Control Guide*.

Maestro has automatic, context-sensitive help (Auto-Help and Balloon Help, or tooltips), and an online help system. To get help, follow the steps below.

- Check the Auto-Help text box, which is located at the foot of the main window. If help is available for the task you are performing, it is automatically displayed there. Auto-Help contains a single line of information. For more detailed information, use the online help.
- If you want information about a GUI element, such as a button or option, there may be Balloon Help for the item. Pause the cursor over the element. If the Balloon Help does not appear, check that Show Balloon Help is selected in the Maestro menu of the main window. If there is Balloon Help for the element, it appears within a few seconds.
- For information about a panel or the tab that is displayed in a panel, click the Help button in the panel, or press F1. The help topic is displayed in your browser.
- For other information in the online help, open the default help topic by choosing Online Help from the Help menu on the main menu bar or by pressing CTRL+H. This topic is displayed in your browser. You can navigate to topics in the navigation bar.

The Help menu also provides access to the manuals (including a full text search), the FAQ pages, the New Features pages, and several other topics.

If you do not find the information you need in the Maestro help system, check the following sources:

- *Maestro User Manual*, for detailed information on using Maestro
- *Maestro Command Reference Manual*, for information on Maestro commands
- *Maestro Overview*, for an overview of the main features of Maestro
- *Maestro Tutorial*, for a tutorial introduction to basic Maestro features
- *Impact User Manual*, for detailed information on basic Impact tasks and panels
- *Impact Command Reference Manual*, for Impact command syntax
- *Strike User Manual*, for detailed information on using Strike

- *Ligand & Structure-Based Descriptors*, for information on using Liaison for descriptors
- Liaison Frequently Asked Questions pages, at https://www.schrodinger.com/Liaison_FAQ.html
- Known Issues pages, available on the Support Center.

The manuals are also available in PDF format from the Schrödinger Support Center. Local copies of the FAQs and Known Issues pages can be viewed by opening the file `Suite_2009_Index.html`, which is in the `docs` directory of the software installation, and following the links to the relevant index pages.

Information on available scripts can be found on the Script Center. Information on available software updates can be obtained by choosing Check for Updates from the Maestro menu.

If you have questions that are not answered from any of the above sources, contact Schrödinger using the information below.

E-mail: help@schrodinger.com
USPS: Schrödinger, 101 SW Main Street, Suite 1300, Portland, OR 97204
Phone: (503) 299-1150
Fax: (503) 299-4532
WWW: http://www.schrodinger.com
FTP: ftp://ftp.schrodinger.com

Generally, e-mail correspondence is best because you can send machine output, if necessary. When sending e-mail messages, please include the following information:

- All relevant user input and machine output
- Liaison purchaser (company, research institution, or individual)
- Primary Liaison user
- Computer platform type
- Operating system with version number
- Liaison version number
- Maestro version number
- mmshare version number

On UNIX you can obtain the machine and system information listed above by entering the following command at a shell prompt:

```
$SCHRODINGER/utilities/postmortem
```

This command generates a file named *username*`-`*host*`-schrodinger.tar.gz`, which you should send to help@schrodinger.com. If you have a job that failed, enter the following command:

```
$SCHRODINGER/utilities/postmortem jobid
```

where *jobid* is the job ID of the failed job, which you can find in the Monitor panel. This command archives job information as well as the machine and system information, and includes input and output files (but not structure files). If you have sensitive data in the job launch directory, you should move those files to another location first. The archive is named *jobid*`-archive.tar.gz`, and should be sent to help@schrodinger.comIf Maestro fails, an error report that contains the relevant information is written to the current working directory. The report is named `maestro_error.txt`, and should be sent to help@schrodinger.com. A message giving the location of this file is written to the terminal window.

More information on the `postmortem` command can be found in Appendix A of the *Job Control Guide*.

Index

120 West 45th Street, 29th Floor
New York, NY 10036

101 SW Main Street, Suite 1300
Portland, OR 97204

8910 University Center Lane, Suite 270
San Diego, CA 92122

Zeppelinstraße 13
81669 München, Germany

Dynamostraße 13
68165 Mannheim, Germany

Quatro House, Frimley Road
Camberley GU16 7ER, United Kingdom

SCHRÖDINGER®